A strategy to support Engineering Education teaching staff monitoring students' learning process: Metacognitive Challenges

Daniela Pedrosa[1], Leonel Morgado[2], José Cravino[3]

[1] CIDTFF – Research Centre on Didactics and Technology in the Education of Trainers, Department of Education and Psychology University of Aveiro, Aveiro, Portugal.
[2] INESC TEC - Institute for Systems and Computer Engineering, Technology and Science, LEAD, & Universidade Aberta, Coimbra, Portugal.
[3] CIDTFF – Research Centre on Didactics and Technology in the Education of Trainers, School of Science and Technology, University of a Trás-os-Montes e Alto Douro, Vila Real.

Email: dpedrosa@ua.pt, Leonel.Morgado@uab.pt, jcravino@utad.pt

DOI: https://doi.org/10.5281/zenodo.7062198

Abstract

It is increasingly required that Engineering Education courses include activities that promote the development of cognitive skills, such as metacognition. However, including such activities is challenging for lecturers, particularly in Distance Learning contexts. It is also complex, when working online, for teaching staff to carry out monitoring of the metacognitive learning processes of students, understand their difficulties, and provide formative feedback.

In this work, we present the design and discussion of a pedagogical strategy: Metacognitive Challenges (MC), which allows lecturers to monitor the evolution of students' perceptions regarding their learning process We discuss how lecturers can use MCs for formative assessment and how to weave this intervention with individual students or groups. The Design Science Research methodology was adopted for the design, implementation, and demonstration of MCs, applied in a Software Engineering course within a distance learning Informatics Engineering undergraduate programme. We exemplify how MCs have the potential to support monitoring of students' cognitive and metacognitive processes and offer a set of guidelines on how the teaching staff can use them.

In future work, we intend to evaluate the effectiveness of MCs in different teaching contexts, and develop technological solutions that facilitate the monitoring process (reduce the time and effort required for analysis of MC content).

Keywords: Metacognition; Cognitive process monitoring; Software Engineering Education; Self-regulation and Co-regulation of Learning.

1 Introduction

Increasingly, Engineering Education is committed to active and situated learning, putting students in contact with real engineering experiences (Wengrowicz, Dori & Dori, 2018). In software engineering, the novice-to-expert transition requires students to develop advanced technical skills, namely: large-scale programming and software development processes (ACM & IEEE, 2016), the ability to think abstractly, and the adoption of cognitive and metacognitive strategies (Garcia, Falkner & Vivian, 2018). Major professional organizations in the field (ACM & IEEE, 2016) recommend connecting practical aspects of real practice with educational plans.

Distance learning in universities has expanded. However, this brings novel challenges and dropout is usually higher (Broadbent, 2017; Pedrosa et al., 2021). Students experience difficulties planning, developing, and using self and co-regulation learning (SCRL) skills properly (ibid.), which adds new challenges to online teaching, both regarding course structure and the practice of e-pedagogy (Kara et al., 2019). Strategies that allow lecturers to overcome such challenges include: 1) provide formative assessment; 2) provide timely, continuous, and constructive feedback that facilitates the process of planning, managing learning and problem-solving skills; 3) adoption of appropriate assessment tools contributing to a better optimization of learning; and 4) reflect upon and adjust their pedagogical practices towards enriching students' learning (Kebritchi, Lipschuetz & Santiague, 2017; Kara et al., 2019).

In Engineering Education, the incorporation of metacognition in curriculum plans is advantageous as it helps students improve essential skills and their ability to engage in SCRL (Wengrowicz, Dori, & Dori, 2018; Wallin & Adawi, 2018). Also, metacognitive regulation is a characteristic that distinguishes experts from novices (Kim & Lim, 2019), hence developed in the novice-to-expert transition.

Metacognition is "*thinking about thinking*" (Flavell, 1979) and is understood as knowledge and capacity for self-judgment about one's own cognitive processes and control, that allows identifying successful strategies (e.g., planning, analysis, management), emotional self-efficacy monitoring, and evaluating metacognitive knowledge according to feedback (Wengrowicz, Dori, & Dori, 2018; Prather et al., 2020; Dindar, Jarvela & Jarvenoja, 2020; Schuster et al., 2020). When the student has metacognitive awareness about these strategies, performance improves and success in higher (Davis & Hadwin, 2021; Frasier, 2021).

Metacognition can be understood in two dimensions (Wengrowicz, Dori, & Dori, 2018): 1) The regulation of cognition - which allows students to develop skills, such as: planning, monitoring and evaluation of their tasks, which allow them to take control of their learning; and 2) Knowledge of cognition – which helps students improve the way they learn and solve problems, in conjunction with the regulation of cognition that allows students to acquire key skills in engineering.

Software Engineering requires problem-solving processes; but it is difficult to study cognitive control (Prather et al., 2020). The use of pedagogical strategies that promote the acquisition and use of metacognitive skills by students is crucial to support teaching feedback processes that encourage self-analysis of errors and positive correction (Sáiz-Manzanares & Montero-García, 2015). However, there are few instructional programs that explicitly focus on teaching control and monitoring skills (Schraw & Gutierrez, 2015) and few studies explore the use of strategies that facilitate formative assessment processes by teaching staff (Wallin & Adawi, 2018).

In this work, we present the design of a strategy that we developed, the Metacognitive Challenges – MC (Pedrosa et al., 2021). We discuss how MC can be used by teaching staff to monitor the cognitive and metacognitive processes of students and to provide formative feedback.

2 Teaching context

In previous work, we explained that the concept the Metacognitive Challenges (MC) results from the articulation of the different dimensions that involve a reflexive metacognition process with the concept of challenges (Pedrosa et al., 2021) and reported on the positive perceptions of students about MC: they are perceived as being innovative, motivational, and useful to develop self-regulating learning strategies and greater self-awareness about one's own skills.

The MC were designed within scope of the e-SimProgramming didactic approach (Pedrosa, 2021). This approach was implemented in an online asynchronous course ("Software Development Laboratory", LDS in the Portuguese-language acronym), part of the 2nd semester of the 2nd year of the Informatics Engineering undergraduate programme at Universidade Aberta (UAb), using the Moodle platform. It is organized along a six-topic syllabus with the goal of scaffolding undergraduates transitioning from novice programmers into proficient programmers that acknowledge the relevance of employing engineering structural qualities in the development of their software programs. The students are typically working students, aged 24-60 years old, residing in various regions of Portugal and abroad, with different academic backgrounds. The teaching and learning methodology employ Project-Based Learning, through the development of software projects by students or teams, integrating concepts sequentially throughout the semester (Pedrosa et al., 2020, 2021).

3 Methodology

This work focuses on the problem that the teaching staff faces when monitoring the evolution of cognitive and metacognitive processes of engineering students, which are essential for successful learning, particularly in Distance Learning. We adopted Design Science Research (Hevner & Chatterjee, 2010), which develops knowledge by embodying it in the design, implementation, and evaluation of an artefact. In this study, we present and demonstrate through examples how Metacognitive Challenges (artefact) can be a strategy that helps teaching staff monitor students' metacognitive processes. We evaluated the examples by collecting data on their use with 32 students (out of 50 enrolled in the LDS) who agreed to participate in this research during the academic year 2019/2020. All students granted their authorization through an informed consent statement.

4 Pedagogical design of Metacognitive Challenges

4.1 Design and implementation:

Two types of Metacognitive Challenges (MC) were designed and implemented (Pedrosa et al., 2021) with different pedagogical goals, timing, and characteristics (see table 1), allowing both students and lecturer to monitor the cognitive and metacognitive learning process of students.

Table 1: Differences between the two types of Metacognitive Challenges (MC)

Dimensions	MC Type 1 Metacognitive challenges as fortnightly reflections about the learning progress	MC Type 2 Metacognitive challenges to promote self-reflection about programming concepts
Pedagogical Goals	Stimulate self-reflection and self-assessment of students about: a) the learning progress. b) the development of the project. c) self-confidence about their work.	Promote students' self-reflection on their ability to apply the knowledge they have acquired throughout the course, regarding the technical aspects of software development processes. Note: For the construction of these MC, in each topic, Bloom's taxonomy was used.
Components of the metacognitive domain	Regulation of cognition: Planning, monitoring (self-awareness), and evaluation (self-assessment).	Knowledge of cognition: Declarative knowledge, Procedural knowledge, and Conditional knowledge. Metacognitive experiences: Feelings and judgments.
When it appears in the course	At the end of each syllabus topic.	At key moments in each syllabus topic (beginning, middle, or near the end).
Number of MC	6	12
Format	Both types of MC have been implemented in Moodle, using the Quiz feature. In the introductory part, a narrative by the fictional Catmming character is used (Pedrosa et al., 2021) that triggers the student's reflection to respond and reflect through prompts (questions in the quiz). The quiz includes Likert-scale closed questions (Very Low; Low; Regular; High; Very High), and open-ended questions where students justify their choice or explain in detail the answer given in the closed question.	
Questions	Standard questions that are adjusted by syllabus topic and the expected development status of the software project.	There are no standard questions. The questions vary according to the metacognitive goals defined for each syllabus topic.
Usefulness for the lecturer	It provides awareness of the student's perception of the learning regulation processes adopted throughout each topic. Identify the type of difficulties; Provide SRL strategies for the student (e.g., planning, time management, seeking help); Checking discrepancies regarding the level of confidence and self-assessment with their appreciation of the work.	It allows the lecturer to infer and formulate formative hypotheses regarding expected skills: whether they were developed, if knowledge was applied correctly, and understand students' feelings and judgments about their technical and knowledge skills.

4.2 Demonstration/Evaluation: How metacognitive challenges help lecturers monitor and be aware of students' metacognitive learning processes?

The Metacognitive Challenges (MC) provide awareness information to the lecturer about the cognitive and metacognitive processes of the students regarding their own learning. With this information the lecturer can define ways of acting/intervene to provide formative feedback for the different situations that occur, in a personalized way (individually) or at the group (class) level.

Ideally, the lecturer should analyse all answers of all students. However, analysing MC is a time-consuming activity. Thus, it is suggested that the lecturer, to gain some awareness about the general panorama of the class, generates response graphs that allow understanding the critical aspects to be solved and define a timeline. We demonstrate and discuss, in the following examples, how to use information from MC to monitor and intervene with feedback.

Example 1 – Motivational intervention

In the question (MC type 1) "*At what level of learning progress do I consider my evolution to be*?", one can see where most students´ answers are. In the following example (figure 1), most students consider that they are at a "Regular" level (satisfactory learning progress).

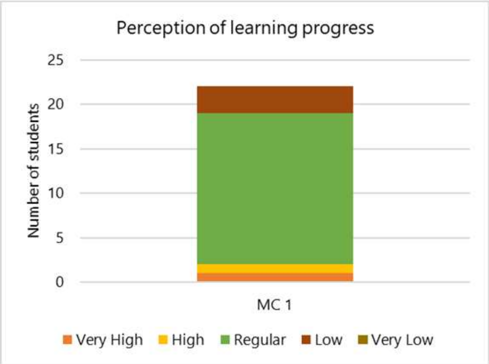

Figure 1: Responses of LDS students for the 2019/2020 academic year to the question "*At what level of learning progress do I consider my evolution to be*?".

In this situation, the lecturer can decide to focus on students who expressed a "Low" level and understand through their answers to the question "*Why?*" the reasons that the students provided. Some examples: "*I started with few bases* (...)". S21, March 25, 2021. "*I think I still have a long way to go.*" S57, March 21, 2021.

Analysing these responses points towards motivational factors and lack of confidence (or modesty). This may inform enable a teaching decision to provide feedback to the whole class of a motivational nature, weaving a set of suggestions for self and co-regulation of learning strategies (such as suggesting that students seek help from the teacher and colleagues). Or instead provide individualized feedback, or yet another approach. The rationale is that the MC provide a structure for teaching decision-making: identifying focus aspects from the Likert-scale responses and then analysing concrete responses within those focus aspects.

Example 2 - Awareness of difficulties and help students overcome them

The lecturer can identify difficulties reported by students and try to understand them. By analysing the answers to the question: "*Did I experience difficulties in this initial phase of software development?*", the lecturer may realize, as in our test case, that most students expressed that they felt difficulties (Figure 2).

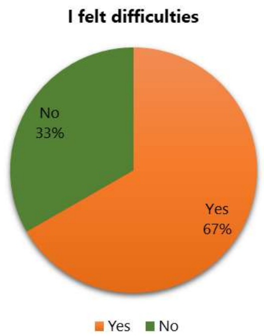

Figure 2: Response of LDS students for the 2019/2020 academic year to the question "*Did I experience difficulties in this initial phase of software development*?" of MC type 1.

The lecturer can try to understand the type of difficulties from the responses to the follow-up question "*Why?*". In our test case, students reported difficulties of various types, e.g.: "*I had a hard time understanding how to use MVC with the Selenium API (...). (...) the interpretation of the texts, the fact that the links to documents were not highlighted, made me start developing the demonstration application when that was not the goal*". S56, April 5, 2020.

In this example, one can identify two types of difficulties:

a) The student's understanding of the practical application of course material (MVC software architecture) with an API (software development tool). The instructional intervention may be to promote class sharing of specific difficulties or offering individualized help, towards better clarification of these concepts to the student or to the class (knowledge of cognition).

b) Understanding specifically which task to perform, from the material provided. In this situation, the lecturer may elect to improve the pedagogical resources for by correcting the next tasks through the placement of visual elements (highlights) and a final task list to help students; or to alert the students in general to check if such misunderstanding occurred to others, etc.

Other responses have shown us that these MC may also expose difficulties not related to the course itself but related to personal lives / professional factors of the students, that affect their self-regulation learning strategies: "*Mainly, personal difficulties led me to compromise time management.*" S36, April 13, 2020. This awareness may enable the teacher to recommend activity prioritization for students, plan recovery plans, or other approaches.

Example 3 - Clarification of aspects related to the syllabus and concepts that may be misunderstood

The question "*Were you able to perform all the requested tasks?"*, enables understanding the students' perception of overall task completion. The lecturer can check whether that perception is correct or not. That is, a student can believe that he/she has completed all tasks, but in reality some task may be missing. This awareness may recommend alerting the student or the class, in realization that uncompleted tasks are not simply delayed, but rather disregarded.

Another approach is this case (figure 3), where most students affirmed that they were able to complete all tasks. If this matches the perception of the lecturer, the focus can shift towards the students who indicated *"No"*, to analyse the reasons.

Again, the "Why" questions will be the source of that analysis. Different types of situations can be encountered requiring interventions, e.g.: "*I await feedback* (...) *to finish this sprint*" S8, April 20, 2020. In this circumstance, there was an expectation of feedback to advance, of which the lecturer might be unaware.

Figure 3: Response of LDS students for the 2019/2020 academic year to the question "*Were you able to perform all the requested tasks?*" of the MC type 1.

Another situation found in this process was related to the student's own technical ability to apply course knowledge in practice: "*I am not able to make user interaction and choice with the API, I changed the model, but it is not easy to be able to make the application according to that design.*" S56, May 9, 2020. In this situation, a technical difficulty was exposed (which may even be impacting other students in the class) and provide the lecturer with an opportunity for general feedback or try to support the student on that specific technical issue (Knowledge of cognition).

Example 4 - Recommendations for adopting learning SCRL strategies

Another MC type 1 question was: *"What are the steps you will take in the next two weeks?"*. This enables the lecturer to verify if students are outlining a plan according to what is expected for the upcoming project phase or deviating from it and may recommend SCRL strategies accordingly (Regulation of cognition).

Example 5 – Motivational or clarification doubts

The level of confidence that the student has in relation to the work developed is an indicator that allows the lecturer to perceive whether the student's confidence (high or low) matches its project status, comparing it to his/her own evaluation. In this example, the answers to the question "*What level of confidence do you feel about the operation of the code of the demonstration application, considering the use of interfaces between components as required*?", exposed that most of students placed their confidence at a "Regular" level (figure 4).

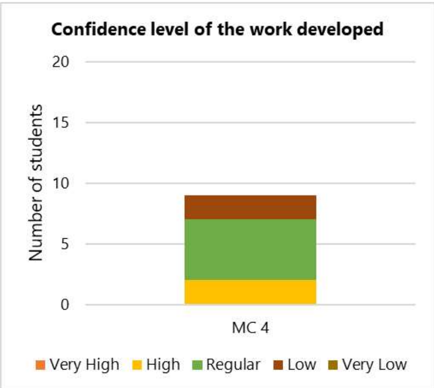

Figure 4: Response of LDS students for the 2019/2020 academic year to the question "*What level of confidence do you feel about the operation of the code of the demonstration application, considering the use of interfaces between components as required*?".

The lecturer may elect to prioritize analysis of student who expressed a "Very Low" or "Low" level of confidence on this aspect. The "why" follow-up question may enable understanding the reasons and allow the lecturer to intervene accordingly. For instance, if the project work matches expectations, perhaps motivational or confidence-building feedback is necessary, e.g.: *"I went back and lowered the level. I still don't understand how the example code works* (…). *I haven't gotten past the classic* [approaches] *yet."* S44, May 10, 2020.

Example 6 - Clarification of assessment criteria

The MC can also enable lecturers to understand whether a student's self-assessment is similar or different from the lecturer's assessment. If the student's self-assessment is very different from the lecturer's, this awareness may originate, for instance, feedback reminding or clarifying the assessment criteria. Comparing the students' confidence level with their self-assessment level may enable the lecturer to focus on discrepant situations, such as someone who has a "Low" level of confidence about the work developed, but self-assessed as "High". For example (table 2).

The lecturer can thus proceed, and possibly explore the reason for this imbalance (regular confidence on a very high self-evaluation), and act in accordance with what is expected for that phase (Knowledge of cognition).

Table 2: Comparison between the student's confidence level and self-assessment regarding an MC.

Student	Confidence level	Why?	Self-evaluation	Why?
S1	Regular	*"I understood the concept of how to implement error handling in the MVC model. I think I could have implemented a more adequate treatment, which I intend to do in the next steps."*	Very High	*"My demo application conforms to the MVC style and is easily adaptable to next steps."*

Example 7 – Improvement for the next editions of the course:

The MC also allow the lecturer to visualize the evolution of the class, in each of the dimensions, verifying which are the critical phases in which he/she will have to act. For example, the lecturer may perceive a reduction in the students' answering of the MC, along with a predominance of students' perceptions of their learning evolution as "Regular". This may anticipate shortcomings that would only be exposed upon later project delivery and enable pre-emptive action. Also, the lecturer may be able to identify phases where the class have felt difficulties, by detecting sudden changes in the responses – this may expose issues with the syllabus, the pedagogical planning, or other transversal problems (figure 5).

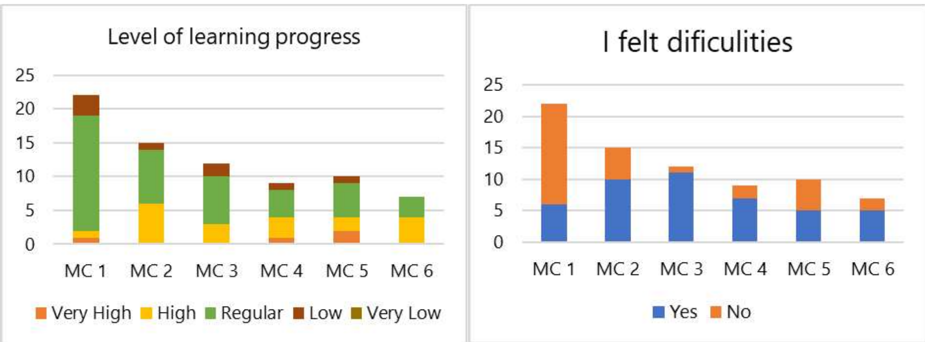

Figure 5: Example of how the MC type1 offer an overview of what happened in the class throughout the course.

The lecturer, when noticing these trends in the answers to the MC, may check for correlations with dropout outcomes (remembering that in Distance Learning dropout rates tend to be high) and intervene. Not just improving the planning/pedagogy in subsequent years, but for the ongoing year, if detected early enough. For instance, acting in a motivational manner, or exploring project development status throughout the class and provide ways to maintain student interest or recover shortcomings. If instead of overall lowering of response rates it is rather a matter of specific students not wanting to address specific MC, the lecturer could try to understand why.

5 Conclusions and final thoughts

The Metacognitive Challenges (MC) emerge as a strategy with the potential for the teaching staff to monitor students' metacognitive learning processes, allowing lecturers to focus their class analysis and feedback effort, and intervene adequately (regulation of cognition and knowledge of cognition) including: 1) Motivational interventions; 2) Recommendations for adopting learning self-regulation and co-regulation strategies; 3) Clarification of doubts due to errors of interpretation, reasoning or resolution; 4) Clarification of assessment criteria; 5) Correction of content or task specification mistakes and/or aspects for pedagogical improvement in subsequent editions of the course (self-reflection of the teacher on his/her pedagogical practices); 6) Understand students' difficulties and help overcome them; 7) Clarification of aspects related to the syllabus and concepts that may be misunderstood.

The awareness of lecturers allows them to make decisions that are critical to the student's learning success. However, analysing MC requires a high amount of effort, so good planning and time management is recommended. Future work will focus on the evaluation of the use of MC by teachers as a form of intervention, verifying their effects on the student's learning process. It is important to develop technological solutions that allow teachers to analyse the contents of MC efficiently and effectively.

6 Acknowledgments

This work was financially supported by National Funds through FCT – Fundação para a Ciência e a Tecnologia, I.P., and CIDTFF (UIDB/00194/2020) - Universidade de Aveiro, Portugal, via Stimulus of Scientific Employment – CEECIND/00986/2017 Individual Support 2017, and via project PTDC/CED-EDG/30040/2017. We would also like to thank all students and teachers who collaborated on this research.

7 References

ACM & IEEE (2016). *Computer Engineering Curricula 2016: Curriculum Guidelines for Undergraduate Degree Programs in Computer Engineering*. ACM & IEEE, New York.

Broadbent, J. (2017). Comparing online and blended learner's self-regulated learning strategies and academic performance. *The Internet and Higher Education*, 33, 24-32. https://doi.org/10.1016/j.iheduc.2017.01.004

Dindar, M., Järvelä, S., & Järvenoja, H. (2020). Interplay of metacognitive experiences and performance in collaborative problem solving. *Computers & Education*, 154, 103922. https://doi.org/10.1016/j.compedu.2020.103922

Flavell, J. H. (1979). Metacognition and Cognitive Monitoring: A New Area of Cognitive-Developmental Inquiry. *American Psychologist* 34 (10): 906–911. https://doi.org/10.1037/0003-066X.34.10.906

Frazier, L. D., Schwartz, B. L., & Metcalfe, J. (2021). The MAPS model of self-regulation: Integrating metacognition, agency, and possible selves. *Metacognition and learning*, 16(2), 297-318. https://doi.org/10.1007/s11409-020-09255-3

Garcia, R., Falkner, K., & Vivian, R. (2018). Systematic literature review: Self-Regulated Learning strategies using e-learning tools for Computer Science. *Computers & Education*, 123, 150-163.

Hevner, A., & Chatterjee, S. (2010). Design science research in information systems. *In Design research in information systems* (pp. 9-22). Springer, Boston, MA.

Kara, M., Erdoğdu, F., Kokoç, M., & Cagiltay, K. (2019). Challenges faced by adult learners in online distance education: A literature review. *Open Praxis*, 11(1), 5-22. https://doi.org/10.5944/openpraxis.11.1.929

Kebritchi, M., Lipschuetz, A., & Santiague, L. (2017). Issues and challenges for teaching successful online courses in higher education: A literature review. *Journal of Educational Technology Systems*, 46(1), 4-29.

Kim, J. Y., & Lim, K. Y. (2019). Promoting learning in online, ill-structured problem solving: The effects of scaffolding type and metacognition level. *Computers & Education*, 138, 116-129. https://doi.org/10.1016/j.compedu.2019.05.001

Pedrosa, D., Fontes, M. M., Araújo, T., Morais, C., Bettencourt, T., Pestana, P. D., Morgado, L., & Cravino, J. (2021). Metacognitive challenges to support self-reflection of students in online Software Engineering Education. *In 2021 4th International Conference of the Portuguese Society for Engineering Education* (CISPEE) (pp. 1-10). IEEE. doi: 10.1109/CISPEE47794.2021.9507230.

Pedrosa, D., Morgado, L., Cravino, J., Fontes, M. M., Castelhano, M., Machado, C., & Curado, E. (2020). Challenges Implementing the SimProgramming Approach in Online Software Engineering Education for Promoting Self and Co-regulation of Learning. In *2020 6th International Conference of the Immersive Learning Research Network* (iLRN) (pp. 236-242). IEEE. https://ieeexplore.ieee.org/document/9155183 .

Prather, J., Becker, B. A., Craig, M., Denny, P., Loksa, D., & Margulieux, L. (2020). What do we think we think we are doing? Metacognition and self-regulation in programming. *In Proceedings of the 2020 International Computing Education Research Conference (ICER '20)*. ACM, New York, NY, USA, 2-13. https://doi.org/10.1145/3372782.3406263

Sáiz-Manzanares, M. C., & Montero-García, E. (2015). Metacognition, self-regulation and assessment in problem-solving processes at university. In *Metacognition: Fundaments, Applications, and Trends* (pp. 107-133). Springer, Cham.

Schraw, G., Gutierrez, A.P. (2015). Metacognitive Strategy Instruction that Highlights the Role of Monitoring and Control Processes. In: *Peña-Ayala, A. (eds) Metacognition: Fundaments, Applications, and Trends. Intelligent Systems Reference Library*, vol 76. Springer, Cham. https://doi.org/10.1007/978-3-319-11062-2_1

Schuster, C., Stebner, F., Leutner, D., & Wirth, J. (2020). Transfer of metacognitive skills in self-regulated learning: an experimental training study. *Metacognition and Learning*, 15(3), 455-477.

Wallin, P., & Adawi, T. (2018). The reflective diary as a method for the formative assessment of self-regulated learning. *European Journal of Engineering Education*, 43(4), 507-521. https://doi.org/10.1080/03043797.2017.1290585

Wengrowicz, N., Dori, Y. J., & Dori, D. (2018). Metacognition and meta-assessment in engineering education. In *Cognition, metacognition, and culture in STEM education* (pp. 191-216). Springer, Cham.